UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/418,323	10/14/1999	MATHIAS LARSSON	LB-2466-41	8745

23117 7590 01/26/2009

NIXON & VANDERHYE, PC
901 NORTH GLEBE ROAD, 11TH FLOOR
ARLINGTON, VA 22203

EXAMINER	
NGUYEN, CHAU T	

ART UNIT	PAPER NUMBER
2176	

MAIL DATE	DELIVERY MODE
01/26/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

RECORD OF ORAL HEARING

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS

AND INTERFERENCES

EX PARTE MATHIAS LARSSON, CHARILAOS CHRISTOPOULOS, MAGNUS JANDEL, DIEGO SANTA CRUZ, and TOURADJ EBRAHMI

Appeal 2008-3118
Application 09/418,323
Technology Center 2100

Oral Hearing Held: December 11, 2008

Before LANCE LEONARD BARRY, ST. JOHN COURTENAY III, and CAROLYN D. THOMAS, *Administrative Patent Judges.*

APPEARANCES:

ON BEHALF OF THE APPELLANTS:

Leo Boutsikaris, Ph.D.
NIXON & VANDERHYE, P.C.
901 North Glebe Road
11th floor
Arlington, Virginia 22203

The above-entitled matter came on for hearing on Monday, December 11, 2008, at The U.S. Patent and Trademark Office, 600 Dulany Street, Alexandria, Virginia, before Janice A. Salas, Notary Public.

THE CLERK: Calendar number 34, Mr. Boutsikaris.

MR. BOUTSIKARIS: Good afternoon. My name is Leo Boutsikaris, and I represent my client today in this appeal hearing. Our position is that the final rejection by the Patent Office is incorrect for the following reasons.

Very briefly explaining our obligation in the embodiments, the method includes -- involves compressing a digital image at the server, storing a compressed version of the image at the server, and then transmitting at least a part of the stored version, compressed version, of the image from the server to the client.

So what happens is this. First, the image is transformed into the frequency domain to yield frequency-domain coefficients. These frequency-domain coefficients are then subdivided into blocks, and then the method compresses the entropy coding, at least the first and the second block of those coefficients.

So we have a set of different, independently decodable coding units, which are stored in the server.

Now, if the server receives a request by the user for a specific part of the image, then the server transmits the coding unit that corresponds to that part of the image without having to employ further entropy encoding for that information.

In other words, the coding units that have been transmitted after the request by the client do not undergo any further processing before they're

transmitted to the client.

Now, the examiner used a combination of two --

JUDGE BARRY: Counsel, before you proceed, let's talk about your invention. The summary of the subject matter wasn't as helpful as it could have been. In fact, actually, I had a lot of problems understanding it, so there's two things I want to focus on.

First of all, it was a good effort that you tried to map things to specific line numbers. That's very helpful.

MR. BOUTSIKARIS: Yes.

JUDGE BARRY: The problem was when I go to the spec, there are no line numbers, and I tried to go ahead and count them myself, but I ended up with lines -- I don't -- in the middle of paragraphs, lines of blank space.

So if you can provide us a numbered copy in a week, by next week --

MR. BOUTSIKARIS: Okay.

JUDGE BARRY: -- that would be great. Provide it specifically to the Board.

MR. BOUTSIKARIS: Okay. Specifying which parts of the claim language correspond to which part of the specification?

JUDGE BARRY: Well, what we really need is a copy of the specification with line numbers.

MR. BOUTSIKARIS: Oh, with line numbers, okay.

JUDGE BARRY: Yes. Because you've already done most of that.

MR. BOUTSIKARIS: Yes. Okay.

JUDGE BARRY: Or a lot of that. Now, what we need you to do here, though, what we really need in these summaries is to take the elements

and map them to the drawings.

MR. BOUTSIKARIS: That's right.

JUDGE BARRY: Now, I couldn't do that because so many of the drawings are so similar, so I'm hoping --

MR. BOUTSIKARIS: I know. Yeah.

JUDGE BARRY: I'm hoping we can -- let's do that now. Let's go ahead and take the two independent claims and do that now. That will save you from having to re -- save you submit it on paper.

So let's go ahead and start with claim 15. With claim 15 about all I can know for sure is where a client is and where a server is, but let's go ahead and start with the first step, transforming the image into a frequency domain. Which elements would you map this to in your drawings? Because there's quite a few client process servers.

MR. BOUTSIKARIS: Well, it's not -- you see, most of drawings have to do with the components that are sent by -- by the client and there's one by the server, so they do not really show their components in the server that does -- that do these kind of transformation.

JUDGE BARRY: So your drawings don't really show what's in, say, most of these claim steps?

MR. BOUTSIKARIS: Hmm --

JUDGE BARRY: Any correspondence you can show would be -- tell us about at this point would be helpful.

MR. BOUTSIKARIS: Okay. I guess figure 4 --

JUDGE BARRY: Okay.

MR. BOUTSIKARIS: In figure 4 there's -- there is a method called

entropy decoding of the sequences.

JUDGE BARRY: Okay. So figure 4 you would say would be your -- would show your first step of transforming the image?

MR. BOUTSIKARIS: Yeah, because it shows that you have a sequence of beads and then you have -- you have -- you start creating the entropy in coding -- entropy coding. So you see that at sequence 3?

JUDGE BARRY: Yes.

MR. BOUTSIKARIS: The transmitter creates masks and the entropy decodes the remaining sequences, okay?

JUDGE BARRY: Okay. So what would be the claim frequency-domain coefficients in figure 4?

MR. BOUTSIKARIS: That would be the content of those blocks because when you think transformation, what you end up is in those frequency coefficients or those coefficients and those are -- these are the boxes.

JUDGE BARRY: Okay. So these are -- the content of sequences, would you say 3, 4, 5?

MR. BOUTSIKARIS: Yeah. In the figure 4, yes. In those boxes, yes.

JUDGE BARRY: Okay. All right. Let's go down to the next step. After transforming, subdividing those coefficients corresponding to the image into at least one block, each block comprising at least one transformed coefficient, where would that be shown in your figures?

MR. BOUTSIKARIS: Again, this is in figure 4. You have -- you have a sequence of different blocks, okay? All right. I think that's the only

place you can have this, like, a schematic of this partitioning into blocks.

JUDGE BARRY: Okay. So the blocks are what the seq -- the blocks -- a sequence is a block?

MR. BOUTSIKARIS: Yeah. You have a sequence of bits, which are really the sequence of coefficients after transformation. Bits correspond to -- after being transformed, correspond to coefficients, so you have a sequence of different coefficients and then you partition those -- that sequence into blocks and then you compress those coefficients using entropy coding.

JUDGE BARRY: So is sequence 4, is that a block?

MR. BOUTSIKARIS: In figure 4?

JUDGE BARRY: Yes.

MR. BOUTSIKARIS: Yeah.

JUDGE BARRY: Okay.

MR. BOUTSIKARIS: Yes. I mean, you see that it says one block says length of remaining entropy coding sequences. I mean, you see that -- so you have -- so this sequence really correspond to the blocks of the frequency coefficients.

JUDGE BARRY: Okay. How about the next step of the -- compressing the entropy encoding, first and a second block into different coding units.

MR. BOUTSIKARIS: Okay. Then it says, ROI encoding is done?

JUDGE BARRY: Mm-hmm.

MR. BOUTSIKARIS: And you see it says coding -- I mean, it codes that specific portion.

JUDGE BARRY: Okay. So that's the R -- that's the compressing.

MR. BOUTSIKARIS: Yes.

JUDGE BARRY: The CUs, are they shown there?

MR. BOUTSIKARIS: The user?

JUDGE BARRY: The decodable coding units.

MR. BOUTSIKARIS: The decodable coding units are what results from this -- after this compression.

JUDGE BARRY: Right. Are they shown here or in one of your other figures?

MR. BOUTSIKARIS: Yeah. All of this is at the end, I guess, at the bottom of figure 4, is that's what is said -- what is being said.

JUDGE BARRY: Okay. So those are the bottom sequences.

MR. BOUTSIKARIS: Yes. Yes.

JUDGE BARRY: All right. How about the next step. Is there a step shown here of storing the coding units on the server in any of the figures?

MR. BOUTSIKARIS: Not explicitly on this figure. I mean, it's -- this is really -- it's stored, and then it's -- well, after the server receives the request, it transmits the appropriate portion.

JUDGE BARRY: Okay. Now, as I look through the figure, there's many places where they mention receiving a request. Which one are you reading this claim step on?

MR. BOUTSIKARIS: Receiving --

JUDGE BARRY: Receiving a request that said server.

MR. BOUTSIKARIS: Right.

JUDGE BARRY: Yeah. Which one is that? 20 --

MR. BOUTSIKARIS: I guess, yes. Yes. Yes, I guess it would be --

that would be in box 3, figure 1. It says, "The client requests a portion of the stored image."

JUDGE BARRY: Okay. Great. So that's step 105.

MR. BOUTSIKARIS: Yes. Step 105, yes.

JUDGE BARRY: Okay. And then the following step?

MR. BOUTSIKARIS: And then the following step is step 107.

JUDGE BARRY: Okay. Great. So that's claim 15. How about -- let's see -- claim 28. Let me see what we need. We have the client-server system. You told us the means for identifying was the client. You said the means for identifying the mask could either be the client or the server.

MR. BOUTSIKARIS: Yes.

JUDGE BARRY: The next means for identifying was the server. In that third element, then, the independently decodable coding units, which ones would those be in figure 4 or would they be in a different figure?

MR. BOUTSIKARIS: Figure 4, that would be the -- the orders that are received at the bottom.

JUDGE BARRY: The bottom ones, okay.

MR. BOUTSIKARIS: Yes, yes.

JUDGE BARRY: Bottom sequences. Means for transmitting. You told us that was the client and you said the request was either 205 or 305. Does that sound right?

MR. BOUTSIKARIS: Yes. 105.

JUDGE BARRY: 105?

MR. BOUTSIKARIS: 105.

JUDGE BARRY: Okay. Okay. Thank you.

MR. BOUTSIKARIS: All right. So the main point here is that the coding units that have been transmitted after the request don't have to undergo any further processing, okay, before they're sent.

Now, the examiner essentially has used two reference. Percival in view of Keith. Percival is related to a method of transforming an image from a server to a client. So it's a similar application.

So according to this gross method, a low detailed version of the image may be sent first and then later transmission of data, more data, can make the image progressively greater in greater detail, okay?

This allows the user, as data's being received, to view a rough version of the image and make a decision whether he wants to see the entire image with better resolution or he wants to see a portion of the image with better resolution.

So this means that the server may receive a request by the client for the following things. Either transmit -- no. Either transmit an entire image until notified otherwise or transmit the entire image but under a predetermined lower resolution.

Third, transmit a portion stamped version of the image. This is a smaller -- the whole image, but a smaller, smaller size. In all -- and this is the bottom most 11 to our claims. Transmit a smaller specific region of the image that is specified by the user. So now, how this is done by Percival.

JUDGE BARRY: Well, counsel, why do you argue -- I'm looking at page 12 of your brief. Why do you argue that in Percival the compression is done after a request is received?

MR. BOUTSIKARIS: Okay. I was going to -- okay. All right.

Okay. As I said, at some point, the client may request a portion of the image. For example, if you look at -- if you look at figure 9B in Percival, the client made a request receipt of that small box 2 in the bottom half -- in the bottom lower end, okay?

So what happens is -- so the client -- the user sends a request for that portion, okay?

Now, when they -- the server receives that request, the way he does it is he transmits different bit-planes in order, okay, and this is done -- and then he says at some point -- Percival says -- then -- Percival says that these bit-planes can be compressed as they're being sent, which means that this happens after the request, okay?

Referring to column 6 --

JUDGE BARRY: Where is the -- where are you reciting?

MR. BOUTSIKARIS: Column 6 in Percival. I'm sorry. No, no, no. I'm sorry. Column 10. Column 10, lines 1 to 6, it says, these bit-planes may be easily compressed by the connects driving continues on and so on.

So what happens is after the survey has received a request to send a specific portion, it starts sending those bit-planes, and then that's when he -- the compression is done.

JUDGE BARRY: Okay.

MR. BOUTSIKARIS: Okay.

JUDGE BARRY: What about a --

MR. BOUTSIKARIS: That's the difference with our -- with our invention where the compression is done before any request is received.

JUDGE BARRY: What about if we look at figure 2 of Percival.

MR. BOUTSIKARIS: Okay.

JUDGE BARRY: The first two blocks.

MR. BOUTSIKARIS: Yes.

JUDGE BARRY: We have a transform in the image is the very first block 60, and then we have the image request coming as the next block 101. Doesn't that show compression being done before the request is received?

MR. BOUTSIKARIS: Okay. Well, again, in Percival, the user's the hard transform.

JUDGE BARRY: Hard transform, correct.

MR. BOUTSIKARIS: If you look at the hard transform on page -- in figure 3, you see it's really reordering of data. It's not compression really.

I mean, look at figure 3. You have -- you get a -- you take a block 68, you've got four pixels and then you rearrange those pixels within a block and then you put back in the -- and you make it with the transformed image, which is the 64, so really you have no compression during the transformation.

And actually -- it was mentioned in the brief -- Percival specifically says that the transformed image has the same memory -- uses the same memory as the original image, so there's no encoding -- no compression done during that transformation.

JUDGE BARRY: Isn't a hard transformed used for image compression?

MR. BOUTSIKARIS: Originally, this hard transform was used here first because it uses distribution of data, reordering of data. You're supposed to send a small portion that has information about the whole image, and then

the client can see that small image and that request maybe more data can be sent to make the clear more clear.

So it's a -- I think the original hard message used is because -- because of this reordering of data, and with different occasions they -- Percival meshes the ordering.

This is what happens. He reorders the data in such a way but you can send progressively different versions of the data image. First, and more often than more are refined, okay? So that's why I don't think it's the -- I don't think it's the compression.

I mean, you can do compression -- as I said, you can do compression after they start sending, after they see the request of a specific portion of the image and then they compress that im -- those bit-planes, but in our claimed invention, the compression's done beforehand. Before any request is received.

So this is the one main difference I think is that Percival teaches the compression is performed after the request is shipped by the user. In ours the compression itself does not compress the image. It just reorders it, okay?

Now, we think that the second difference is this. After the -- in Percival, after the server receives a request, then its further processing is performed, okay, and I will direct you to the section in column 10 of Percival under the title -- under the dynamic or definition of the image field.

And there you see that and then you look up -- and then column 11, basically it says that after the server receives a request for a specific portion of the image, than he goes and performs the -- the --

JUDGE BARRY: Where are you reading, counsel?

MR. BOUTSIKARIS: Yeah. Column 11.

JUDGE BARRY: Right.

MR. BOUTSIKARIS: Yes. The server performs the big shuffling steps in hard transformation.

JUDGE BARRY: What line number is this?

MR. BOUTSIKARIS: Yes. Line 11.

JUDGE BARRY: Good shuffling steps?

MR. BOUTSIKARIS: Yes. So in order for the server to retrieve that specific portion that the client wants and send it, he needs to perform those shuffling steps, the big shuffling steps, okay, and then -- and then he transmits the information, the data.

Again, as opposed to our invention where nothing is done. After the request is received, those blocks, which are already compressed, they have been sent, okay, so clearly in Percival the server does additional steps to send that specific portion of the image.

JUDGE COURTENAY: But does it do entropy encoding? The negative limitation in your claim recites without the server having to employ further entropy encoding with respect thereto.

MR. BOUTSIKARIS: Right.

JUDGE COURTENAY: I'm looking at the last line of claim 15.

MR. BOUTSIKARIS: Right. Yeah, right. And as we said, Judge -- as we just said, Percival doesn't teach entropy coding. Any compression. Okay. That's just reordering of data.

To make it easier to send rough versions of data, of images, to the client, and the client receives -- looks at the rough images and then says, I

want a better image or version or I want a specific part of the version.

JUDGE BARRY: What about claim 2 of Percival, what the examiner cited.

MR. BOUTSIKARIS: Yes. But that refers to the fact that those bit-planes have been sent -- are sent compressed. But that's after the request.

JUDGE BARRY: After the request. I see. Okay.

MR. BOUTSIKARIS: And then also I would like to comment on the combination of the two references.

JUDGE BARRY: Counsel, we're out of time.

MR. BOUTSIKARIS: Okay.

JUDGE BARRY: Thank you.

MR. BOUTSIKARIS: Thank you. So you like us to send -- send you this -- the specification with lines -- with numbers.

JUDGE BARRY: Please. By next Thursday.

(Whereupon, the proceedings were concluded on December 11, 2008.)